

Vilniaus universitetas
Matematikos ir informatikos fakultetas

Kursas: "Programų sistemų inžinerija II"
Laboratorinių darbų reikalavimai

Dėstytoja A. Lupeikienė
audrone.lupeikiene@mii.vu.lt

Vilnius, 2018

1. Bendrieji reikalavimai

1.1. Darbų sąrašas

Eil. Nr.	Darbo pavadinimas	Pateikti iki:	Apginti iki.
1.	Pirmasis laboratorinis darbas	2018-03-12	2018-03-14/19
2.	Antrasis laboratorinis darbas	2018-04-16	2018-04-18/23
3.	Trečias laboratorinis darbas	2018-05-24	2018-05-31

Pastaba. Datos gali šiek tiek slinkti atsižvelgiant į tai, kelis praktinius užsiėmimus turite tą savaitę.

1.2. Laboratorinių darbų atlikimo tvarka

Studentai susiskirsto į pogrupius po 4 žmones. Atsakingas asmuo vadovauja darbo darymui, tačiau jiems pavestas darbo dalis atlieka visi grupės nariai. Konkretus kiekvieno studento indėlis į darbą privalo būti aprašytas to darbo anotacijoje.

Kiekvienas pogrupis gauna kito pogrupio praeitą semestrą sukurta programų sistemos projektinę dokumentaciją ir eksperimentinę realizaciją. Sistema vystoma toliau. Spręsdamas šį uždavinį pogrupis rengia tris dokumentus. Kiekvienas dokumentas pristatomas ir įvertinamas pažymiu. Pirmasis ir antrasis pristatymas (gynimas) vykdomas žodžiu, dalyvauja visi pogrupio nariai. trečiasis pristatymas – tai pranešimas naudojant jam parengtas skaidres. Pristatymo trukmė apie 10 minučių. Bet kurio darbo gynimui skiriama ne mažiau 30 min. Pristato visi ir klausimai gali būti pateikti visiems grupės nariams. Pažymys už gebėjimą pristatyti ir atsakymus į klausimus rašomas kiekvienam pristatančiajam individualiai (nuo 0 iki 1 balo). Pats darbas vertinamas, atsižvelgiant į jo esmę, apipavidalinimą pagal pateiktus reikalavimus, teisingą profesinę terminiją, taisyklingą lietuvių kalbą. Už darbą visi grupės nariai gauna tą patį pažymį (nuo 0 iki 9 balų). Už negebėjimą tinkamai organizuoti darbą ir dirbti pagal numatytą planą **pažymys sumažinamas** 1 balu kiekvieną vėlesnę savaitę. Išimtyt daromos tik ligos ar kitais ypatingais atvejais. Laboratorinių darbų neapgynusiems studentams **egzamino laikyti neleidžiama**.

Darbą (**tekstinį dokumentą arba PDF formatu**) būtina atsiųsti dėstytojui. Siunčiamo teksto **negalima archyvuoti**. Laiško apraše (subject) **reikia nurodyti kursą, grupę, pogrupio numerį ir darbo numerį**. Kartu su trečiojo darbo projekto dokumentacija kartu turi būti pateikti sukurto ir/arba modifikuoto programų pradiniai tekstai.

Pateikiant/ginant darbą, atspausdinta jo kopija atiduodama dėstytojui.

Semestro metu bus atliekamos 4 individualios užduotys - kontroliniai.

Konsultacijos teikiamos ir darbai pristatomi laboratoriniams darbams numatytu laiku.

2. Pirmojo laboratorinio darbo „Reikalavimų apibrėžimas“ struktūra

2.1. Reikalavimai

Šiame skyriuje pateikiami reikalavimai sistemai, apibrėžiantys norimą sistemos pokytį. Reikalavimai turi apimti tiek funkcinis, tiek nefunkcinis programų sistemos aspektus, jie turi būti identifikuojami. ICONIX procese daroma prielaida, kad pradiniai reikalavimai sistemai yra netikslūs, nevienareikšmiški, galbūt prieštaringi, nepilni ir t. t. Tad, darbo pradžioje šiame skyriuje turi būti pateikti pirminiai reikalavimai, kurie darbo eigoje tikslinami ir gerinama specifikacijos kokybė. Atsiskaitant pirmą laboratorinį darbą šie reikalavimai turi būti suderinti su struktūriniu dalykinės srities modeliu ir sistemos užduotimis.

[Skyrelyje pateikiama](#)

- 1) Reikalavimai, nusakantys norimą sistemos pokytį. Jie *gali būti* pateikti kaip atspindintys naudotojo istoriją. Pvz.

Pavadinimas : Klientas grąžina prekę		
Priėmimo testas: custRetItem	Prioritetas: 1	Sąnaudos : 2
Kai klientas grąžina prekę, ji turi būti autentifikuota. Jei pirkinys yra autentiškas, į pirkėjo sąskaitą turi būti pervesti pinigai (ne ilgiau kaip per 10 sekundžių) arba pirkinys keičiamas kitu. Atitinkamai turi būti modifikuota sandėlio informacija.		

Tipinis vartotojo istorijų formulavimo šablonas būna toks:

As a student I want to purchase a parking pass so that I can drive to school.

As a consumer, I want shopping cart functionality to easily purchase items online.

As an executive, I want to generate a report to understand which departments need to improve their productivity.

2) Pakeisti programų sistemos reikalavimai, t. y. modifikuota ankstesnė reikalavimų specifikacija. Taip pat pateikiami pakeitimai, kurie turi būti matomi išreikštiniu pavidalu. Tai galima padaryti vienu iš dviejų būdų.

a) Reikalavimų specifikacijoje skirtingai pažymimas pradinis ir pakeistas tekstas. b) Pildoma reikalavimų pakeitimų lentelė:

Data	Pradinis reikalavimas	Klaidos aprašas	Pakeitimo tipas	Naujas reikalavimas
				(arba jo identifikatorius reikalavimų specifikacijoje)

Pakeitimo tipas:

A – pridėtas, D – pašalintas, M – modifikuotas.

Pastaba. Pildant reikalavimų pakeitimų lentelę nereikia pamiršti, kad reikalavimų specifikacijos ji nepakeičia.

2.2. Struktūrinis dalykinės srities modelis

Šiame skyriuje pateikiamas struktūrinis nagrinėjamos dalykinės srities modelis. Modelis pateikiamas UML klasių diagramomis kartu su žodynu - sąrašu esybių su jų aprašymais. Jei klasės grupuojamos į paketus, turi būti pateikta ir atitinkama paketų diagrama. Darbo pradžioje struktūrinis dalykinės srities modelis apibrėžiamas, nurodant tik klases ir jų ryšius. Apibrėžiant užduotis šis modelis patikslinamas bei papildomas atributais. Skyriaus pabaigoje pateikiama reikalavimų – struktūrinio dalykinės srities modelio atsekamumo matrica.

[Skyrelyje pateikiama](#)

1) Dalykinės srities esybių sąrašas, sudarytas nagrinėjant reikalavimų specifikaciją (joje išskiriant daiktavardžius, įvardžiutinius būdvardžius).

2) Pradinė klasių diagrama, gauta praeitame žingsnyje sudaryto dalykinės srities esybių sąrašo pagrindu (klasės ir jų ryšiai).

3) Jei klasės grupuojamos į paketus, pateikiama paketų diagrama.

4) Reikalavimų – struktūrinio dalykinės srities modelio atsekamumo matrica. Ji turi tokį pavidalą:

	1 klasė	2 klasė	...
1 reikalavimas		√	
2 reikalavimas	√		
...			

2.3. Užduotys

Šiame skyriuje pateikiamos sistemos atliekamos užduotys. Kiekviena jų turi turėti pavadinimą, pagrindinį ir alternatyvius scenarijus. Jei reikia, su užduotimis turi būti pateikti ir naudotojo sąsajos eskizai ar kitos priemonės, naudotos užduočių apibrėžimo ir tikslinimo procese. Užduotims aprašyti turi būti taikomos UML užduočių (angl. use case) ir paketų diagramos. Pabrėžtina, kad šiame skyriuje užduočių tekstas yra svarbesnis nei diagramų turinys.

Pateikiant darbą vertinimui, užduotys jau turi būti patikslintos, peržiūrėtas jų atitikimas reikalavimams ir struktūriniam dalykinės srities modeliui (atlikta preliminarinė peržiūra). Be užduočių, skyriaus pabaigoje turi būti pateikta ir reikalavimų - užduočių atsekamumo matrica.

Skyrelyje pateikiama

- 1) Pradinė užduočių diagrama, sudaryta nagrinėjant reikalavimų specifikaciją.
- 2) Naudotojo-sistemos sąveikos scenarijai kiekvienai užduočiai. Tai tekstas, susidedantis iš pagrindinio ir alternatyvaus scenarijaus aprašo. Pageidautina, kad kartu būtų sudarytos ir tinkamos UML diagramos (paprastai tai sekų diagramos).
- 3) Jei reikia, interfeisų prototipai, grafinės iliustracijos ar ekranų modeliai, kurie buvo naudoti užduotims detalizuoti. (Sąveika vyksta per ekranus, langus, puslapius ar pan.)
- 4) Jei užduotys grupuojamos į paketus, pateikiama užduočių paketų diagrama.
- 5) Papildyta (apimant reikiamus ryšius) ir patikslinta užduočių diagrama kiekvienam užduočių paketui. Taip pat atitinkamai pataisyti ir užduočių aprašai.

Užduočių aprašai gali būti sudaryti iš šių dalių:

Užduoties pavadinimas.

Užduoties vykdymą inicijuojantis įvykis (trigeris).

Pirminis aktorius.

Antrinis aktorius.

Užduoties vykdymo prieš-sąlygos.

Pagrindinis scenarijus.

Prieš-sąlyga.

Scenarijaus aprašas.

Alternatyvūs scenarijai.

Prieš-sąlyga-1.

Scenarijaus-1-aprašas.

...

Prieš-sąlyga-n.

Scenarijaus-n-aprašas.

Pastaba. Gali būti panaudoti ir kitokie tinkami užduočių aprašymo šablonai.

Use Case Name	Make Fuel Delivery.
Purpose	To allow a paying customer to deliver fuel of a customer selected grade.
Preconditions	The desired fuel grade is available.
Invariants	Tank level > 4% tank capacity while pump is turned on.
Primary Scenario	1. Customer removes gun from holster; 2. Attendant enables pump; 3. Customer selects one fuel grade; 4. Pump motor is started; 5. Customer uses trigger to control fuel delivery, causing the system to engage/disengage pump clutch as trigger is depressed/released; 6. Customer replaces gun in holster; Pump motor is stopped.
Postconditions	At least two litres of fuel have been delivered. The pump gun is in the holster. The pump motor is off.
Secondary Scenario 1	Customer delivers less than two litres of fuel.
Postconditions	Less than two litres of fuel have been delivered.
Secondary Scenario 2	Tank level falls below 4%..
Postconditions	All pumps connected to tank are disabled..
Secondary Scenario 3	Customer releases trigger but does not return gun to holster.
Postconditions	Pump switched off after timeout period..
Performance Specification	
Trigger	Customer removes gun from holster.
Periodicity	Aperiodic.
Rate	60/hour.

2.4. Reikalavimų specifikacijos, dalykinės srities modelio ir užduočių diagramos (apimant tekstinius aprašus) peržiūros rezultatai

(Peržiūra daroma grupės nariams pasiskirsčius skirtingus naudotojų ir sistemos kūrėjų vaidmenis).

[Skyrelyje pateikiama](#)

- 1) Pataisyta reikalavimų specifikacija. (Su užregistruotais pokyčiais, apimant rastas klaidas.)
- 2) Pataisyta klasių diagrama. (Su visais reikiama ryšiais.)
- 3) Pataisyta užduočių diagrama. (Aprašuose naudojami tik dalykinės srities elementai, įvardinti ekranai, pažymėtos planuojamos realizuoti užduotys.)

4) Reikalavimų – užduočių atsekamumo matrica. Ji turi tokį pavidalą:

	1 užduotis	2 užduotis	...
1 reikalavimas	√		
2 reikalavimas	√	√	
...			

3. Antrojo laboratorinio darbo „Reikalavimų analizė ir techninė architektūra“ struktūra

3.1. Robustiškumo analizė (kitai, eskizinis projektavimas)

[Skyrelyje pateikiama](#)

1) Kiekvienai užduočiai (angl. use case) nubraižyta robustiškumo diagrama. Kartu pateikiamas užduočių aprašas. (Užduočių aprašai apima: **Užduoties pavadinimas, Pagrindinis scenarijus, Alternatyvūs scenarijai.**) Jei užduočių aprašas jau buvo padarytas pirmame darbe, čia turi būti pateiktas pakoreguotas, jei to reikia.

2) Papildyta ir patikslinta klasių diagrama (pridėtos naujos klasės ir klasių atributai).

3) Jei reikia, taisoma reikalavimų specifikacija, neužmirštant registruoti pokyčius ir pakomentuoti klaidas.

3.2. Eskizinio projektavimo peržiūra.

Tikslas – įsitikinti robustiškumo diagramų teisingumu ir kad robustiškumo diagramos, struktūrinis dalykinės srities modelis bei užduočių (angl. use-case) tekstiniai aprašai yra išsamūs, suderinti ir juose nėra prieštaravimų.

[Skyrelyje pateikiama](#)

1) Pataisytos robustiškumo diagramos. Kartu pateikiamas pataisytos užduočių diagramos ir aprašai.

2) Pataisyta klasių diagrama.

3) Jei reikia, taisoma reikalavimų specifikacija, neužmirštant registruoti pokyčius ir pakomentuoti klaidas. Jei manote reikalinga, gali būti pateiktos atsekamumo matricos.

3.3. Techninė architektūra.

[Skyrelyje pateikiama](#)

1) Kuriamos sistemos architektūra (tai gali būti aprašyta schema).

2) Sistemos dalių išdėstymo (angl. deployment) diagrama.

Pridedamas aprašas, kuriama pagrindžiami priimti sprendimai.

4. Trečiojo laboratorinio darbo „Projektavimas ir įgyvendinimas“ struktūra

Darant šį darbą siūloma toliau tęsti sistemos projektavimą, t. y. pirmiausia sudaryti sekų diagramas kiekvienai užduočiai. Po to, pabaigti detalizuoti klasių diagramą, papildant trūkstamais atributais ir pridedant operacijas. Gautą projektinę dokumentaciją apjungti su pirmuose darbuose sudaryta dokumentacija. Toliau nurodytas rekomenduojamas darbo turinys šiai medžiagai sudėti.

4.1. Sistemos detalusis projektas

[Skyrelyje pateikiama](#)

4.1.1. Sistemos užduotys

Užduočių diagrama su užduočių vykdymo scenarijais (galutinė pataisyta).

Robastiškumo diagramos (galutinės pataisytos).

Sekų diagramos (po vieną kiekvienai užduočiai, galutinės pataisytos).

4.1.2. Sistemos struktūra

Klasių diagrama (galutinė pataisyta, su atributais ir operacijomis).

4.2. Detalaus projektavimo peržiūra

[Skyrelyje pateikiama](#)

4.2.1. Peržiūra

Trumpas komentaras, kaip buvo vykdoma detalaus projekto peržiūra.

Peržiūros rezultatai: rastos klaidos ar pakeisti projektiniai sprendimai, kartu trumpai argumentuojant naujus sprendinius.

4.2.2. Atsekamumas

Reikalavimų - klasių ir reikalavimų - užduočių atsekamumo matricos (pataisytos).

4.3. Testavimo planas ir scenarijai

[Skyrelyje pateikiama](#)

4.3.1. Programinių vienetų testai.

4.3.2. Sistemos užduočių testai.

4.4. Sistemos techninė architektūra.

[Skyrelyje pateikiama](#)

4.4.1. Sistemos komponentų diagrama (galutinė pataisyta).

4.4.2. Išdėstymo diagrama (galutinė pataisyta)

4.5. Sistemos realizacija

[Skyrelyje pateikiama](#)

4.5.1. Duomenų bazės schema.

4.5.2. Pradiniai programų kodai (sukurti arba modifikuoti) ir aprašas, pagrindžiantis priimtus sprendimus.

Priedai

1. Priedas. Pataisyta reikalavimų specifikacija.

2. Priedas. Pateikiama kita jums atrodanti reikalinga medžiaga.

1 priedas. Titulinio lapo pavyzdys

VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS

„ELEKTRONINIO MOKYKLOS DIENYNO SISTEMA“
(E-dienynas)

Verslo tikslų ir poreikių specifikacija

(1 versija)

Darbą atliko 3 kurso IN 2/1 grupės 4: pogrupio studentai
Vismantas Pavardenis (darbo vadovas)
Edvardas Pavardenis
Leukadija Pavardenytė
Rokas Pavardenis

Vilnius, 2018

2 priedas. Darbo anotacijos pavyzdys

V. Pavardenis, E. Pavardenis, L. Pavardenytė, R. Pavardenis. Elektroninio mokyklos dienyno sistema. Verslo tikslų ir poreikių specifikacija (1 versija). VU PS katedra, Vilnius, 2009. – 18 p., iliustr.

Šiame darbe pateiktas kurso “Programų sistemų inžinerijos įvadas” laboratorinis darbas, skirtas verslo tikslų ir poreikių specifikavimui. Tai pirmasis iš keturių pagal šį kursą daromų laboratorinių darbų. Darbas skirtas išsiaiškinti užsakovo verslo problemas, nutarti, ar jas spręsti gali padėti kokia nors programų sistema, ką tokia sistema turėtų daryti, kaip, ją sukūrus, ja būtų naudojama ir kokią konkrečią naudą ji duotų. Darbe atlikta išorinė mokymo proceso analizė, nustatytos problemos, su kuriomis susiduria mokykla mokymo procese, potencialios grėsmės bei neišnaudotos mokymo proceso galimybės. Jame taip pat aptartos vidinės mokymo proceso problemų bei grėsmių priežastys, pasiūlyta šio proceso tobulinimo strategija, ji išreikšta strateginių bei operacinių tikslų medžiu ir išanalizuota, kokias informacines ir skaičiuojamąsias paslaugas turėtų teikti šią strategiją palaikanti programų sistema. Pasiūlyti tokios programų sistemos naudojimo scenarijus ir veiksmų planas jam įgyvendinti, atlikta sistemos įgyvendinamumo analizė, o taip pat parodyta, koku mastu kokias mokymo proceso užduotis sistema padės spręsti ir kokią konkrečią naudą ji teiks mokytojams, moksleiviams bei jų tėvams.

Informacija apie vykdytojus ir jų įnašą į darbą:

Vismantas Pavardenis (darbo vadovas, vismantas@pastas.lt): verslo proceso analizė, sistemos teikiamos naudos analizė, išvados.

Edvardas Pavardenis (edvardas@pastas.lt): sistemos įgyvendinamumo analizė, sistemos naudojimo analizė.

Leukadija Pavardenytė (leukadija@pastas.lt): UML diagramos, verslo proceso analizė, sistemos naudojimo analizė.

Rokas Pavardenis (rokas@pastas.lt): sistemos įgyvendinamumo ir teikiamos naudos analizė, verslo proceso analizė.

3 priedas. Specifikacijos pavyzdys (papasakojimas)

Petrol Station Control System Requirements

A computer-based system is required to control the dispensing of petrol, to handle customer payment and to monitor tank levels. The system must be “best of breed”, easy to use, reliable, fast and easy to modify to incorporate requirements yet to be conceived.

Before a customer can use the self-service pumps, the pump must be enabled by the attendant. When a pump is enabled, the pump motor is started, if it is not already on, with the pump clutch free. When the trigger in the gun is depressed, closing a micro switch, the clutch is engaged and petrol pumped. When it is released, the clutch is freed. There is also a micro switch on the holster in which the gun is kept which prevents petrol being pumped until the gun is taken out. Once the gun is replaced in the holster, the delivery is deemed to be completed and the pump disabled. Further depressions of the trigger in the gun cannot dispense more petrol. After a short standby period, the pump motor will be turned off unless the pump is re-enabled.

A metering device in the petrol line sends a pulse to the system for each 1/100 litre dispensed. The cost of the fuel is calculated using the amount delivered and unit cost which is displayed on the pump.

There are two kinds of pump. The normal kind allows the user to dispense petrol ad lib. The sophisticated pumps, imported from New Zealand, allow the customer to preset an amount and a volume of petrol. Petrol will then be pumped up to a maximum of the required quantity.

Transactions are stored until the customer pays. Payment may be either in cash, by credit card or account. A customer may request a receipt (showing price, time, date etc.) and will get a token for every 5 pounds. Customers sometimes abscond without paying and the operator must annotate the transaction with any available information, the vehicle’s registration number for example. At the end of the day, transactions are archived and may be used for ad hoc enquiries on sales.

At present, two grades of petrol are dispensed from five pumps on the forecourt. Each pump takes its supply from one of two tanks, one tank for each grade. The tank level must not drop below 4% of the tank’s capacity. If this happens, the pumps serviced by that tank cannot be enabled to dispense petrol.

The developer shall provide a complete software system to perform the above control functions and shall provide design documents, code listings and so on.

4 priedas. Specifikacijos pavyzdys (reikalavimų formulavimas)

- 9.1.2 System shall provide food for pets.
 - 9.1.2.1 System shall accept user input for intervals to deliver food.
 - 9.1.2.2 System shall notify users when food in storage reaches low levels, as users will be required to fill the storage depot.
 - 9.1.2.3 Pet's food shall be delivered to their bowls every user-defined interval.
 - 9.1.2.4 System shall allow user to set portion weight for every bowl for the system.
 - 9.1.2.5 Pet food delivery shall be portioned to user-defined weight.

- 9.2.1 Coffee maker shall be tied into the water purification system.
- 9.2.2 System shall start coffee maker at any user-defined time as long as water is present, coffee bean levels are sufficient, and unit is powered.
- 9.2.3 System shall send a notification when bean levels are low.
- 9.2.4 When bean levels are too low to make coffee, system shall send an alert and coffee maker shall blink a warning indicator.

- 9.12.1 System shall serve as answering machine for household.
- 9.12.2 System shall allow users to configure number of rings before answering.
- 9.12.3 System shall allow users to configure any number of phone mail boxes for recipients.
- 9.12.3 System shall allow users to record greeting message that will be played after user-defined number of rings.
- 9.12.4 System shall allow users to configure greeting message to be played for individual mail boxes.
- 9.12.5 System shall record messages for recipients along with date, time stamp, and incoming phone number to non-volatile memory.
- 9.12.6 System shall send out notification to users when they have a new message in their mailbox (i.e., email, text message, pages, etc.).